

DynamoDB

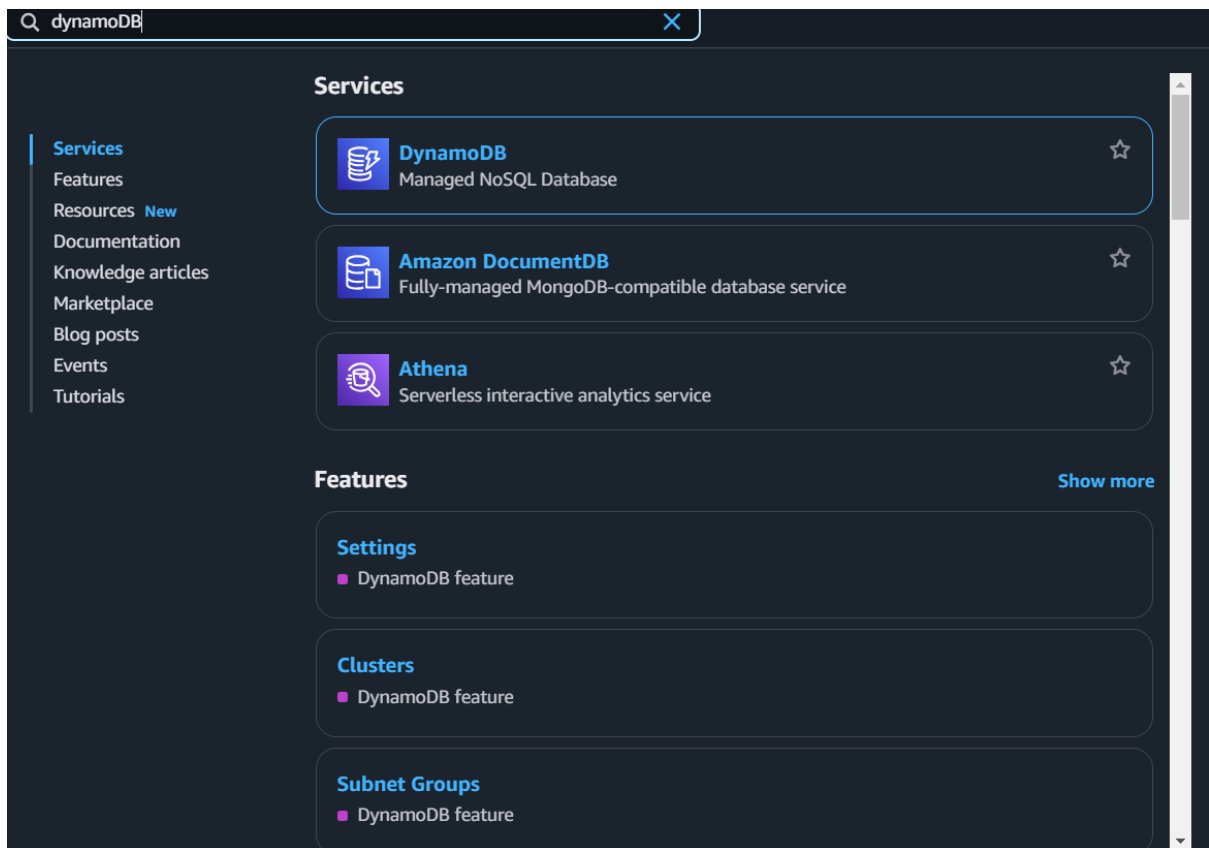
DynamoDB is a fully managed NoSQL database service offered by Amazon Web Services (AWS). It is designed for applications requiring high performance at any scale. DynamoDB is serverless, meaning it automatically scales up or down to adjust for capacity and maintains high availability without the need for infrastructure management.

Key Features of DynamoDB

1. **Scalability:** Handles large amounts of data and traffic effortlessly.
2. **NoSQL:** Key-Value and Document data models for flexibility.
3. **Serverless:** AWS manages the infrastructure for you.
4. **High Availability:** Provides fault-tolerant replication across multiple availability zones.
5. **Performance:** Low-latency responses with automatic scaling.
6. **Security:** Offers encryption at rest, fine-grained access control, and AWS Identity and Access Management (IAM) integration.

Here, are visual Steps following are :

➤ **Steps 1:**



❖ Steps 2:

[Alt+S]

[DynamoDB](#) > [Tables](#) > Create table

Create table

Table details [Info](#)

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name
This will be used to identify your table.

Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.).

❖ Step : 3

Partition key
The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

String ▼

1 to 255 characters and case sensitive.

Sort key - optional
You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

String ▼

1 to 255 characters and case sensitive.

❖ Step 4:

Table settings

☒ **Default settings**
The fastest way to create your table. You can modify most of these settings after your table has been created. To modify these settings now, choose 'Customize settings'.

☐ **Customize settings**
Use these advanced features to make DynamoDB work better for your needs.

❖ Step 5:

Default table settings
These are the default settings for your new table. You can change some of these settings after creating the table.

Setting	Value	Editable after creation
Table class	DynamoDB Standard	Yes
Capacity mode	On-demand	Yes
Maximum read capacity units	-	Yes
Maximum write capacity units	-	Yes
Local secondary indexes	-	No
Global secondary indexes	-	Yes
Encryption key management	Owned by Amazon DynamoDB	Yes
Deletion protection	Off	Yes
Resource-based policy	Not active	Yes

❖ Step 6:

Tags
Tags are pairs of keys and optional values, that you can assign to AWS resources. You can use tags to control access to your resources or track your AWS spending.
No tags are associated with the resource.

Add new tag

You can add 50 more tags.

❖ Step 7:

The Class_10 table was created successfully.

Tables (1) Info

Find tables

Any tag key

Any tag value

< 1 >

Name ▲

Status ▼

Partition key ▼

Sort key ▼

Indexes ▼

Replication Regions ▼

Deletion protection ▼

Favorite ▼

Read capacity mode ▼

Write capacity

Class_10

Active

Roll No (S)

-

0

0

Off

☆

On-demand

On-demand

❖ Step 8:

Create item

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. [Learn more](#)

FormJSON view

AttributesAdd new attribute

Attribute name	Value	Type	
Roll No - Partition key	104	String	
Name	Shruti	String	Remove
Age	18	Number	Remove

CancelCreate item

❖ Step 9:

Items returned (1/4)

ActionsCreate item

< 1 > ⚙️

	Roll No (String)	Age	Name
<input type="checkbox"/>	104	18	Shruti
<input type="checkbox"/>	103	20	Shyam
<input type="checkbox"/>	102	24	Aman
<input checked="" type="checkbox"/>	101		